

## **BAB II LANDASAN TEORI**

### **2.1 Sistem**

Dalam mendefinisikan sistem dapat menggunakan dua kelompok pendekatan sistem yaitu menekankan pada prosedur dan menekankan pada komponen atau elemennya.

Menurut Jerry Fitz Gerald, dan kawan-kawan dalam Jogiyanto HM (2000, Hal 1-2) berpandangan bahwa sistem merupakan jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

Definisi ini akan mempunyai peranan yang penting di dalam pendekatan untuk mempelajari suatu sistem. Secara sederhana, suatu system dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel-variabel yang terorganisir, saling berinteraksi, saling ketergantungan satu sama lainnya dan terpadu.

Menurut Abdul Kadir (2003, hal 11) dalam buku Pengenalan Sistem Informasi, sistem terbentuk dari sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan.

Komponen atau elemen merupakan subsistem-subsistem dalam suatu sistem yang tidak dapat berdiri lepas sendiri-sendiri namun akan saling membentuk suatu kesatuan.

### 2.1.1 Karakteristik Sistem

Model umum sebuah system adalah *input*, proses dan *output* yang merupakan konsep sebuah sistem sederhana. Sebuah system memiliki karakteristik atau sifat tertentu. Karakteristik yang dimaksud adalah sebagai berikut (Jogiyanto HM, 2000, Hal 3):

a. **Komponen Sistem**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

b. **Batasan Sistem (*Boundary*).**

Batas sistem merupakan daerah yang membatasi antara suatu system dengan sistem yang lainnya atau dengan lingkungan luarnya atau pemisah antara system dan daerah diluar system.

c. **Lingkungan Luar Sistem (*Enviroment*).**

Lingkungan luar dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.

d. **Penghubung Sistem (*Interface*).**

Penghubung merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke sistem lain

e. Masukkan Sistem (*Input*).

Masukkan adalah energi yang dimasukkan ke dalam sistem atau segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk di proses .

f. Keluaran Sistem (*Output*).

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan bisa juga dikatakan hasil dari suatu pemrosesan.

g. Pengolahan Sistem (*Process*).

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah masukan menjadi keluaran atau proses juga merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna.

h. Sasaran Sistem (*Objective*)

Suatu sistem pasti mempunyai tujuan (*goal*) dan sasaran (*objective*). Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dalam keluaran yang dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran dan tujuannya.

### 2.1.2 Klasifikasi Sistem

Terdapat empat Klasifikasi sistem dalam beberapa sudut pandang yaitu ((Abdul Kadir 2003, hal 11) dan ((Jogiyanto HM, 2000, Hal 3):



a. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik dapat berupa gagasan atau konsep. Sistem fisik adalah sistem yang ada secara fisik dan dapat oleh mata kita.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia.

c. Sistem tertentu dan sistem tidak tentu

Sistem tertentu adalah sistem yang operasinya sudah dapat diprediksi. Sistem tidak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

d. Sistem tertutup dan sistem terbuka

Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

## 2.2 Pengertian Informasi

### a. Data

Data dalam sudut pandang Jogiyanto (2000, Hal 8) adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata". Sedangkan kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu. Sedangkan Abdul Kadir (2003, Hal 29) mengatakan bahwa data merupakan deskripsi tentang benda, kejadian, aktivitas, dan transaksi, yang tidak mempunyai makna atau tidak berpengaruh langsung kepada pemakai

### b. Informasi

Jogiyanto (2005, Hal 8) mendefinisikan bahwa Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sedangkan dilihat dari kaca mata McFadden dkk (2000) dalam Abdul Kadir (2003, Hal 31) informasi sebagai data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut.

## 2.3 Pengertian Sistem Informasi

Robert A. Leitch dan K. Roscoe Davis dalam Jogiyanto (2005, Hal 11) mendefinisikan bahwa sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Hall (2001) mengatakan system informasi adalah sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi dan distribusikan kepada pemakai dalam buku Abdul Kadir (2003, hal 11).

John dan Gary Grudnitski mengemukakan sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan. Blok bangunan tersebut yaitu (Jogiyanto, 2005, Hal 11) :

a) Blok Masukan (*input block*)

Blok masukan mewakili data yang masuk ke dalam sistem informasi yang termasuk di dalamnya metode-metode dan media untuk menangkap data yang dapat berupa dokumen-dokumen dasar.

b) Blok Model

Blok model terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data.

c) Blok Keluaran

Produk keluaran sistem informasi adalah informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen.

d) Blok Teknologi

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasil dan mengirimkan keluaran dan membantu pengendalian dari sistem keseluruhan.



e) Blok Basis Data

Basis data adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan dalam perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

f) Blok Kendali

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal yang dapat merusak sistem dapat dicegah ataupun apabila terdapat kesalahan maka akan cepat terselesaikan.

## 2.4 Internet

*Internet (International Network)* merupakan suatu jaringan yang memiliki jangkauan mencakup seluruh dunia sehingga memungkinkan sebuah komputer untuk terhubung dengan komputer lain yang jaraknya mungkin beribu-ribu kilometer.

*Internet* terdiri dari gabungan *Local Area Network (LAN)* dan *Wide Area Network (WAN)*.

## 2.5 E-Commerce

*E-commerce* secara umum dapat diartikan sebagai proses transaksi jual beli secara elektronik melalui media internet. Menurut Mariza Arfina dan Robert Marpaung *e-commerce* atau yang lebih dikenal dengan *e-com* dapat diartikan sebagai suatu cara berbelanja atau berdagang secara *online* atau *direct selling* yang memanfaatkan fasilitas Internet dimana terdapat *website* yang dapat menyediakan layanan "*get and deliver*".

Menurut **David Baum**, pengertian *e-commerce* adalah: “*E-Commerce is a dynamic set of technologies, applications, and business process that link enterprise, consumers, and communities through electronic transactions and the electronic exchange of goods, services, and information*”. *E-Commerce* merupakan satu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang dilakukan secara elektronik (David Baum dalam Onno W. Purbo, 2000 : 2).

Kalakota dan Whinston(2005) mendefinisikan *e-commerce* dari beberapa perspektif antara lain :

a. Komunikasi

*E-commerce* merupakan pengiriman barang, servis. Informasi, atau pembayaran melalui jaringan komputer atau dengan alat komputer lainnya.

b. Proses Bisnis

*E-commerce* merupakan suatu teknologi aplikasi untuk otomatisasi dari transaksi bisnis dan arus kerja.

c. Servis

*E-commerce* merupakan suatu alat yang menunjukkan keinginan dari perusahaan, konsumen dan manajemen untuk mengurangi biaya servis sementara meningkatkan kualitas barang dan kecepatan layanan pengiriman.



d. Online

E-commerce menyediakan kemampuan untuk melakukan pembelian dan penjualan produk dan informasi di *Internet* atau layanan *online* lainnya.

Ada tiga tipe utama *e-commerce* yaitu:

a. *Business-to-Constumer* (B2C)

*Business-to-Constumer* merupakan transaksi online dibuat antara bisnis dan konsumen. Transaksi ini meliputi transaksi penjualan dengan pembili-pembeli individu

b. Business-to- Business (B2B)

Didalam B2B taransaksi *online* dibuat antara bisnis dan bisnis yang lain, partisipannya meliputi bisnis atau organisasi yang lain

c. Business-to-Employee (B2E)

Didalam *Business-to-Employee* transaksi online dibuat antara bisnis dan pegawai atau karyawan. Transaksi yang ada hanya berupa informasi atau servis untuk kepentingan bisnid perusahaan.

*E-Commerce* berdasarkan pada derajat digitalisasi dari produk, proses dan pengiriman mempunyai dua macam bentuk, yaitu:

a. Pure E-Commerce yaitu dimensinya digital

b. Partial E-Commerce semua kemungkinan-kemungkinan lainnya, mempunyai campuran dari dimensi fisik dan dimensi digital

## 2.6 Unified Modelling Language

Menurut Munawar (2005, hal 17) *Unified Modelling language* (UML) adalah suatu alat bantu yang sangat handal di dunia pengembangan sistem yang

berorientasi objek, UML menyediakan pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lain.

Romi Satria Wahono dan Sri Dharyanti *Unified Modelling Language* merupakan sebuah bahasa yang telah menjadi bahan standar dalam dunia industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Dengan menggunakan UML, dapat membuat model untuk semua jenis aplikasi peranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, dan dapat digunakan juga untuk mendefinisikan notasi dan *syntax*. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram-diagram piranti lunak, setiap bentuk mempunyai makna tertentu dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan.

Uml menyediakan beberapa notasi dan *artifact* standar yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. *Artifact* didalam UML didefinisikan sebagai informasi dalam bentuk yang digunakan untuk atau dihasilkan dalam proses pengembangan perangkat.

Ada tiga modeling yang terdapat dalam UML yaitu :

- a. *Function Modelling* merupakan diagram-diagram yang menyatakan bagaimana suatu sistem itu bekerja dan yang termasuk dalam *function modeling* adalah activity diagram, usecase deskripsi dan use case diagram.

- b. *Struktural Modelling* terdiri dari beberapa diagram yang memberikan *snapshot* dari elemen-elemen yang berperilaku. *Struktural modeling* mempunyai beberapa model diantaranya *Class Diagram* dan *Class Responsibility Colaboration* (CRC Card).
- c. *Behavior Modelling* mendefinisikan bagaimana suatu objek bertindak dan bereaksi, dan berhubungan dengan fungsi diterapkan pada suatu atribut. Mempunyai beberapa model yaitu *sequence digram* dan *state machine diagram*.





<i>Major Area</i>	<i>View</i>	<i>Diagrams</i>	<i>Konsep Utama</i>	
<i>Struktural</i>	<i>Static view</i>	<i>Class diagram</i>	<i>class, association,</i>	
			<i>generalization, dependency</i>	
			<i>realization, interface</i>	
	<i>use case view</i>	<i>use case diagram</i>	<i>usecase actor, association,</i>	
			<i>extend, include, usecase</i>	
			<i>generalization</i>	
	<i>implementation</i>	<i>component diagram</i>	<i>component, interface, dependency</i>	
			<i>realization</i>	
	<i>deployment</i>	<i>deployment</i>	<i>node, component, dependency,</i>	
<i>Diagram</i>				
<i>location</i>				
<i>dynamic</i>	<i>state machine</i>	<i>statechart diagram</i>	<i>state, event, transition, action</i>	
			<i>View</i>	
	<i>activity view</i>	<i>activity diagram</i>	<i>state, activity, complexation</i>	
			<i>transition, fork, join</i>	
	<i>interaction</i>	<i>sequence diagram</i>	<i>interaction, object, message,</i>	
			<i>activation</i>	
			<i>collaboration diagram</i>	<i>collaboration, interaction</i>
				<i>collaboration role</i>
				<i>message</i>
	<i>model</i>	<i>model</i>	<i>class diagram</i>	<i>package, subsystem, model</i>
<i>management</i>	<i>management</i>			
	<i>View</i>			
<i>extesibility</i>	<i>All</i>	<i>All</i>	<i>constraint, stereotype</i>	
			<i>tagged values</i>	

**Tabel 2.1 Konsep Dasar Unified Modelling Language.**

## 2.6.1 Function Modelling

### 2.6.1.1 Use Case Description

*Use case* Deskripsi merupakan penjabaran tekstual tentang kejadian dan bagaimana pengguna berinteraksi dengan sistem, *Use case* deskripsi juga berfungsi sebagai ilustrasi narasi dalam menjelaskan permasalahan

### 2.6.1.2 Use case Diagram

Munawar (2005, hal 109-100) menjelaskan *Use case diagram* merupakan deskripsi fungsi dari sebuah sistem perspektif pengguna. *Use case* bekerja dengan cara memdeskripsikan tipikal interaksi antar *user* (pengguna) sebuah sistem dengan sistem-nya sendiri bagaimana sistem tersebut digunakan atau “apa” yang diperbuat sistem dan bukan “bagaimana”. *Use case* dapat dipresentasikan sebuah interaksi antara *actor* dengan sistem .

*Use case Diagram* dapat sangat membantu bila sedang menyusun *requirement* (Jacobsoonet all, 2000). Diagram ini menunjukkan tiga aspek dari sistem yaitu *actor*, *use case*, dan sistem atau subsistem.

Dilihat dari segi pandang Romi Satria Wahono dan Sri Dharyanti sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend* *use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

## 2.6.2 Struktural Modelling

### 2.6.2.1 Class Diagram

*Class* dilihat dari segi pandang Romi Satria Wahono dan Sri Dharyanti merupakan sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah



objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan. satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain.

*Class* memiliki tiga area pokok :

1. Nama (dan *stereotype*)
2. Atribut
3. Metoda

Atribut dan metoda dapat memiliki salah satu sifat berikut :

- *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan
- *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya
- *Public*, dapat dipanggil oleh siapa saja

*Class* dapat merupakan implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metoda. *Interface* tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *class*. Dengan demikian *interface* mendukung resolusi metoda pada saat *run-time*.

### **Hubungan Antar Class**

1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah *navigability* menunjukkan arah *query* antar *class*.



2. Agregasi, yaitu hubungan yang menyatakan bagian.
3. Pewarisan, yaitu hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan (message) yang di-passing dari satu class kepada class lain. Hubungan dinamis dapat digambarkan dengan menggunakan sequence diagram yang akan dijelaskan kemudian.

#### **2.6.2.2 CRC Card (Class Responsibility Colaboration)**

Munawar (2005, hal 98) berpendapat bahwa *Class Responsibility Colaboration Card*(CRC Card) adalah mengidentifikasi tanggung jawab yang harus dilakukan oleh sebuah objek, aksi yang harus dilakukan oleh objek, atau beberapa pengambilan oleh objek dan ini dibutuhkan oleh class lain untuk bekerja sama. Hal ini memberikan ide tentang link diantara *class-class* meskipun masih dilevel tinggi.

Salah satu manfaat dari CRC Card adalah mendorong adanya diskusi diantara pengembang. Dengan CRC card pemodelan *interaction* bisa dilakukan secara cepat, karena kartu-kartu tersebut bisa dipindahkan secara cepat untuk mendapat nilai *alternative* lain

maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun *run time*. Umumnya komponen terbentuk dari beberapa *class* dan/atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

### 2.6.3.3 Deployment Diagram

*Deployment/physical diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisik. Sebuah *node* adalah server, *workstation*, atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya. Hubungan antar *node* dan *requirement* dapat juga didefinisikan dalam diagram ini.

## 2.7 Bahasa Pemrograman PHP (*Personal Home Page*)

PHP merupakan sebuah bahasa scripting yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat.

Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi **.html**. File html ini dikirimkan oleh server (atau file) ke browser, kemudian browser menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php,



program ini harus diterjemahkan oleh web-server sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut.

Program PHP dapat ditambahkan dengan mengapit program tersebut di antara tanda `<? dan ?>`. Tanda-tanda tersebut biasanya disebut tanda untuk escaping (kabur) dari kode html. File html yang telah dibubuhi program php harus diganti ekstensi-nya menjadi **.php3 atau .php**. PHP merupakan bahasa pemrograman web yang bersifat server-side *HTML=embedded scripting*, di mana script-nya menyatu dengan HTML dan berada di server. Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di server tetapi disertakan HTML biasa. PHP dikenal sebagai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis seperti ASP (*Active Server Pages*) dan JSP (*Java Server Pages*).

## 2.8 Database MySQL

MySQL adalah sebuah sistem manajemen database relasi (relational database management system) yang bersifat “terbuka” (open source). Terbuka maksudnya adalah MySQL boleh di download oleh siapa saja. Baik versi kode program aslinya (source code program) maupun versi binernya (executable program) dan bisa digunakan secara (relatif) gratis baik untuk dimodifikasi sesuai dengan kebutuhan seseorang maupun sebagai suatu program aplikasi komputer. MySQL menggunakan bahasa standar SQL (Structure Query Language) sebagai



bahasa interaktif dalam mengolah data. Perintah SQL juga sering disebut Query. Karena menggunakan bahasa standar yang sama maka tidak akan menjadi kendala besar bila suatu saat nanti berhubungan dengan database selain MySQL. Tentu saja tetap ada perbedaan sintaks bahasa paket program SQL tersebut. Adapun beberapa kelebihan MySQL adalah sebagai berikut :

- a. **Kecepatan.** Dengan adanya fitur tambahan Query Chacing pada MySQL rilis 4.0, kinerja query secara umum akan naik rata-rata 200% dari kinerja biasanya.
- b. **Open source.** Dengan konsep ini siapapun dapat berpartisipasi mengembangkan MySQL dan hasil penembangan itu diserahkan kepada umum atau kepada komunitas Open Source.
- c. **Kapabilitas.** MySQL telah digunakan untuk mengelola database dengan jumlah 50 juta record. Bahkan sanggup untuk mengelola 60.000 tabel dengan jumlah baris 5.000.000.000. Mendukung penggunaan index hingga 32 index pertabelnya, sedangkan setiap indexnya terdiri dari 1 hingga 16 kolom kriteria. Informasi ini dapat di peroleh dari manual MySQL yang didapat di download dari situs MySQL.
- d. **Konektifitas dan keamanan.** MySQL mendukung dan menerapkan sistem keamanan dan ijin akses tingkat lanjut, termasuk dukungan keamanan dengan pengacakan lapisan data (SSL transport layer encryption). Bahkan pada MySQL rilis 4.0 dapat diterapkan pembatasan penggunaan sumberdaya server berdasarkan penggunaan per user. Adanya tingkatan user dan jenis akses yang beragam dapat memberikan batasan akses berdasarkan lokasi pengakses, dan terdapatnya jaminan keamanan pada MySQL.

- e. **Fleksibilitas dan Portabilitas.** MySQL mendukung perintah-perintah ANSI SQL 99 dan beberapa perintah database alternative lainnya sehingga mempermudah untuk beralih dari dan ke MySQL.
- f. **Lintas platform sistem operasi.** MySQL dapat dijalankan pada beberapa sistem operasi yang berbeda, seperti Linux, Microsoft Windows, FreeBSD, Sun Solaris, Mac OS, HP-UX, AIX, QNX, Novell NetWare, SCO OpenUnix, SGI Iric, dan Dec OSF.

## 2.9 XAMPP

Xampp merupakan sebuah tool yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi web server Apache, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi. Software XAMPP terdiri atas:

### a. APACHE

Tugas utama apache adalah menghasilkan halaman web yang benar kepada peminta, berdasarkan kode PHP yang dituliskan oleh pembuat halaman web. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu database diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman web yang dihasilkan.

### b. PHP

Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat web yang bersifat server-side scripting. PHP memungkinkan



kita untuk membuat halaman web yang bersifat dinamis. PHP dapat dijalankan pada berbagai macam Operating System (OS), misalnya Windows, Linux dan Mac OS. Selain Apache, PHP juga mendukung beberapa web server lain, misalnya Microsoft IIS, Caudium, PWS dan lain-lain. Sistem manajemen database yang sering digunakan bersama PHP adalah MySQL. Namun PHP juga mendukung system manajemen Database Oracle, Microsoft Acces, Interbase, d-Base, PostgreSQL dan sebagainya.

PHP mendukung penuh Object Oriented Programing (OOP), integrasi XML, mendukung semua ekstensi terbaru MySQL, pengembangan web services dengan SOAP dan REST, serta ratusan kemampuan. Sama dengan web server lainnya PHP juga bersifat open source sehingga setiap orang dapat menggunakannya dengan gratis.

c. MySQL

MySQL dapat digunakan untuk membuat dan mengola database beserta isinya. Dengan MySQL dapat digunakan untuk menambahkan, mengubah dan menghapus data yang berada dalam database. MySQL merupakan sistem manajemen database yang bersifat relational. Artinya data-data yang dikelola dalam database akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan menjadi jauh lebih cepat.

MySQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar. MySQL juga dapat

menjalankan perintah-perintah Structured Query Language (SQL) untuk mengelola database-database yang ada di dalamnya.

d. PHPMyAdmin

Pengelolaan database dengan MYSQL harus dilakukan dengan mengetikkan baris-baris perintah yang sesuai (command line) untuk setiap maksud tertentu. Jika ingin membuat database, ketikkan baris perintah yang sesuai untuk membuat database. Jika ingin menghapus tabel, ketikkan baris perintah yang sesuai untuk menghapus tabel. Hal tersebut tentu cukup menyulitkan karena harus hafal dan mengetikkan perintahnya satu persatu. Banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola data base dalam MySQL, salah satunya adalah phpMyAdmin. Dengan phpMyAdmin kita dapat membuat tabel, mengisi data dan lain-lain dengan mudah tanpa harus hafal perintahnya. Untuk mengaktifkan phpMyAdmin langkah-langkahnya adalah : yang pertama setelah XAMPP diinstall, kemudian aktifkan web server Apache dan MySQL dari control panel XAMPP. Yang kedua, jalankan browser (IE) lalu mengetikkan alamat web berikut : <http://localhost/phpmyadmin/> pada address bar lalu tekan Enter. Langkah ketiga apabila telah nampak interface (tampilan antar muka) phpMyAdmin, bisa memulainya dengan mengetikkan nama database, nama tabel dan seterusnya.



## 2.6.3 Behavior Modelling

### 2.6.3.1 Sequence Diagram

*Sequence diagram* dilihat dari kacamata Romi Satria Wahono dan Sri Dharyanti menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

Munawar mengatakan *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.

Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message*. Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan *icon* khusus untuk objek *boundary*, *controller* dan *persistent entity*.

### 2.6.3.2 Component Diagram

*Component diagram* menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code*